

REMARKS

Claims 1-10, 12, 13, and 15-28 are pending. Claims 1, 5, 6, 12, 13, 15, 22, 24, and 28 are amended. Claims 10, 19, and 23 are canceled. Claims 29 and 30 are new. Reconsideration and reexamination are respectfully requested in view of the following remarks.

I. The § 112 Rejections

The Examiner has rejected claims 12 and 15 under 35 U.S.C. § 112 as being indefinite. Claims 12 and 15 have been amended as suggested by the Examiner to overcome this rejection and are now in condition for allowance.

II. The § 103 Rejections

The Examiner has rejected claims 1-10, 13, 17-20, 22-23, 26-27 under 35 U.S.C. § 103(a) as being unpatentable over Mapedit Imagemap Editing Software, Version 2.3 for Windows 3.1, 1997 by Boutell.Com, Inc. URL: <http://www.boutell.com/mapedit>, pp.1-19 ("Mapedit").

Claim 1, as amended, recites a method for receiving from a user an input selecting a layer in an electronic artwork having a plurality of layers, the selected layer having content consisting of one or more non-transparent regions in combination having a boundary in a transparent frame; defining an area based on the boundary; assigning an action to the area; and associating the area and the action with the selected layer as a property of the selected layer in the electronic artwork. The Examiner concedes that Mapedit does not teach inputting a graphic file containing layers but states that "Mapedit teaches the saving of edited overlapping layered image regions" (Mapedit, Figures 17-19). The applicant disagrees, and disagrees that in any event this would disclose or suggest the method recited in amended claim 1.

Mapedit appears to disclose a software application for creating and editing an "image map" (see e.g. Figure 1 ("Mapedit Imagemap Editing Software") and Figure 8). As the reference explains, "mapedit allows you to load your image into a scrollable, resizable window and then draw polygons, circles and rectangles on top of it, specifying at URL for each" (Figure 8). Thus Mapedit appears to disclose an application in which the user defines hotspots in an image map by first loading a pre-existing image into an editing window, manually drawing polygons in the

editing window, and associating URLs with the polygons. The reference also states that “when hotspots overlap, the oldest gets the click” (see Figure 12). This appears to refer to the instance in which the user draws multiple, overlapping polygons in the editing window, such as is shown in Figures 17 and 18. Turning to the rejection, these manually created overlapping rectangles appear to be what the Examiner refers to as “layered image regions”.

Mapedit's overlapping polygon image maps do not amount to “an electronic work having a plurality of layers” as amended claim 1 requires. The applicant's specification makes it clear that “layer” means an image layer used in a graphics application to build a final image by compositing several image layers together, and explains that such image layers typically include image data, an optional mask and compositing controls (Specification, page 4, lines 18-19, 25-26). Nothing in the Mapedit reference discloses or suggests that any of the disclosed polygons are composited to build a final image. Instead, the Mapedit polygons are added manually by the user to define one or more hotspots in an image map, and do not appear to contribute to the underlying image in any way. Accordingly, because Mapedit does not address images having multiple layers as that term is used in the applicant's specification, the reference cannot disclose or suggest receiving a user input selecting such a layer.

This becomes even more clear in claim 1's recitation that the selected layer must have content consisting of one or more non-transparent regions in a transparent frame. Even assuming, as the Examiner appears to, that the overlapping rectangles in Figures 17 and 18 constitute such layers (in the form of “overlapping image regions”), Figures 17 and 18 clearly show that the portions of the underlying image that fall within the overlapping rectangles is fully visible and not obscured. Thus, the “layers” in question are *transparent*, and contain no “non-transparent regions” as claim 1 requires.

Another significant difference between Mapedit and the method recited in claim 1 is how hotspot areas are defined. Claim 1 recites that an area is defined based on the boundary of the combined regions in the layer. An action is then assigned to the area, thereby creating a hotspot. The action and area are associated with the selected layer as a property of the layer in the electronic artwork.

By contrast, Mapedit discloses that to define hotspots (for an image having a single image layer as contrasted to a plurality of layers as discussed above), the user must manually draw the boundary of the area, thereby selecting the shape of the hotspot, its location within the single image layer and its dimensions. Thus, in Mapedit, hotspots are defined based on manual user input - the drawings of one or more polygons in an editing window - and not based on the boundary of combined non-transparent regions in a selected layer, as claim 1 requires.

The applicant respectfully submits that for at least the aforementioned reasons claim 1 would not have been obvious in view of the cited references. The applicant further submits that claims 2-4, 12, 13, 15-18 and 29, which depend from claim 1 also would not have been obvious for at least the same reasons discussed above.

Claim 13 is independently allowable over the Mapedit reference for at least the following additional reason. Claim 13 recites the method of claim 1, further comprising calculating any dynamic content for the selected layer before the area is defined. The specification defines "dynamic content" as data that is computed from other data at the time the layers are composited (Specification, page 5, lines 3-5). The Examiner states, "Since it is known in the art that currently edited information is considered dynamic information until saved, Mapedit's calculation and formulation of hotspots is based upon dynamic content, prior to saving." Even assuming that "currently edited information" can be considered dynamic, it is not immediately apparent how Mapedit's manual definition of hotspot polygons discloses "calculating dynamic content for the selected layer before the area is defined" as claim 1 requires. But more importantly, the rejection appears to ignore that "dynamic content" is given a specific definition in the specification - that is content that is *computed from other data* at the time the layers are composited. Mapedit merely discloses a method of adding hotspots to a static image and does not appear to even suggest any way to associate a hotspot with an image layer having dynamic content as that term is defined in the specification.

Independent claim 5 recites a computer program product having similar features to claim 1. It and dependent claims 6-9, 20-22, and 24-27, which depend from it, are allowable for at least the reasons discussed above in the context of claim 1.

Claims 12 and 21 are also rejected as being unpatentable over Mapedit in view of U.S. Patent No. 6,034,689 ("White"). Claim 12 recites the method of claim 1, further comprising conforming the hotspot area automatically to content of the selected layer when the electronic artwork is edited. White merely discloses *rescaling* an image map area when a web page is resized to fit different display areas, not changing the hotspot area to conform to content of a layer when the underlying image is edited. Accordingly, the applicant respectfully submits that all of the elements recited in claims 12 and 21 are not taught by the references and the claims are patentable over Mapedit in view of White.

Claims 15-16, and 24-26 are rejected as being unpatentable over Mapedit in view of U.S. Patent No. 5,991,781 ("Nielsen"). The applicant respectfully submits the Examiner has misconstrued Nielsen. Claim 15 teaches the method of claim 1 wherein the selected layer has two or more *non-contiguous* non-transparent regions, which in combination define the area, which is associated with an action. By contrast, Nielsen, teaches having more than one *contiguous* regions and having a different action associated with each contiguous region. Accordingly, the applicant respectfully submits that all of the elements recited in claims 15-16, 24-26 are not taught by the references and the claims are patentable over Mapedit in view of Nielsen.

Finally, claim 28 is rejected as being unpatentable over Mapedit in view of U.S. Patent No. 5,956,710 ("Habermehl"). The Examiner states that Habermehl discloses "defining hot spot areas taking into account the boundaries of an area, said defining accomplished via neural net." Claim 28 recites the method of claim 1, wherein defining the area comprises converting each non-transparent region to a perimeter boundary path and fitting a user-selected shape to the perimeter boundary path, wherein the shape defines the area. By contrast, Habermehl teaches a method whereby "the user defines the specified region by selecting points within the region by performing an act such as randomly clicking an input device such as a mouse, associated with a cursor, within the specified region" (Col. 3, lines 22-25). Habermehl clearly does not teach a user-selected shape being fitted to the perimeter boundary path of a non-transparent region. Accordingly, the Applicants submit claim 28 is allowable over Mapedit in view of Habermehl.


Applicant : Michaud et al.
Serial No. : 09/058,496
Filed : April 10, 1998
Page : 10

Attorney's Pocket No.: 07844-273001 / P247

Attached is a marked-up version of the changes being made by the current amendment. Applicant asks that all claims be allowed. Applicant believes that no fees are due. However, the Commissioner is authorized to apply any charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: 4/30/07



Timothy A. Porter
Reg. No. 41,258

Fish & Richardson P.C.
2200 Sand Hill Road, Suite 100
Menlo Park, CA 94025
Telephone: (650) 322-5070
Facsimile: (650) 854-0875



RECEIVED

MAY 14 2001

Technology Center 2100

Version with markings to show changes made

In the claims:

Claims 10, 19, and 23 are canceled.

Claims 29 and 30 are new.

Claims 1, 5, 6, 12, 13, 15, 22, 24, and 28 have been amended as follows:

1. (Amended four times) A method comprising:

receiving from a user an input selecting a layer in an electronic artwork having a plurality of layers, the selected layer having content consisting of one or more non-transparent regions in a transparent frame;

for the selected layer of the artwork, [calculating a definition of] defining an area based on [corresponding to] a boundary of the one or more non-transparent regions in combination; [, and]

assigning an action to the area, the action defining a function that is to be activated when the area is selected; and

associating the area and the action with the selected layer as a property of the selected layer in the electronic artwork.

5. (Amended twice) A computer program, tangibly stored on a computer-readable medium, comprising instructions for causing a computer to:

receive an electronic artwork having a plurality of layers, each layer having transparency information [;] defining one or more non-transparent regions in the layer in a transparent frame;

receive from a user an input selecting one of the plurality of layers;

[identify a non-transparent region of the selected layer of the electronic artwork;]

for the selected layer of the artwork, define an area based on a boundary of the one or more non-transparent regions in combination; and

assign an action to [an] the area, [corresponding to the non-transparent region,] the action defining a function that will be activated when the area is selected.

6. (Amended) The computer program of claim 5, further comprising instructions to:
convert the non-transparent region into a perimeter boundary path; and
fit a user-selected shape to the perimeter boundary path, wherein the shape defines the
area [calculate a boundary of the non-transparent region; and
calculate a definition of the area from the boundary.]
10. [The method of claim 1, further comprising:
receiving from a user of a graphics application operating on the electronic artwork an
input that selects the layer.]
12. (Amended twice) The method of claim 1 [11], further comprising:
conforming the area automatically to content of the selected layer when the electronic
artwork is edited. [by recalculating the definition of the area when the content of the selected
layer changes.]
13. (Amended) In a graphics application that supports dynamic content in layers, the method of
claim 1, further comprising:
calculating any dynamic content for the selected layer before the area is [calculated]
defined.
15. (Amended) The method of claim 1 [14], wherein:
the selected layer has two or more non-contiguous non-transparent regions in a
transparent frame; and
the area defined is based on the boundary of the non-transparent regions in combination.
19. [The computer program of claim 5, further comprising instructions for causing a computer
to:
receive from a user an input that selects the layer of the electronic artwork.]

22. (Amended) The computer program of claim 5, further comprising instructions for causing a computer to:

calculate any dynamic content for the selected layer before the area is [calculated]
defined.

23. [The computer program of claim 5, wherein:

the layer has one or more non-transparent regions in a transparent frame; and
the non-transparent region or regions in combination define the area.]

24. (Amended) The computer program of claim [23] 5, wherein:

the layer has two or more non-contiguous non-transparent regions in a transparent frame;
and

the area is defined based on the boundary of the non-transparent regions in combination
[define the area].

28. (Amended) The method of claim 1, wherein:

[calculating a definition of] defining the area comprises converting each non-transparent region to a perimeter boundary path and fitting a user-selected shape to the perimeter boundary path, wherein the shape defines the area. [by tracing the outer boundary of each non-transparent region in the layer.]

29. (New) The method of claim 3, further comprising:

outputting the composited artwork as an image file; and
outputting an HTML file including an image map for the area and a URL for the action.

30. (New) The computer program of claim 5, wherein the action is a URL (Uniform Resource Locator).